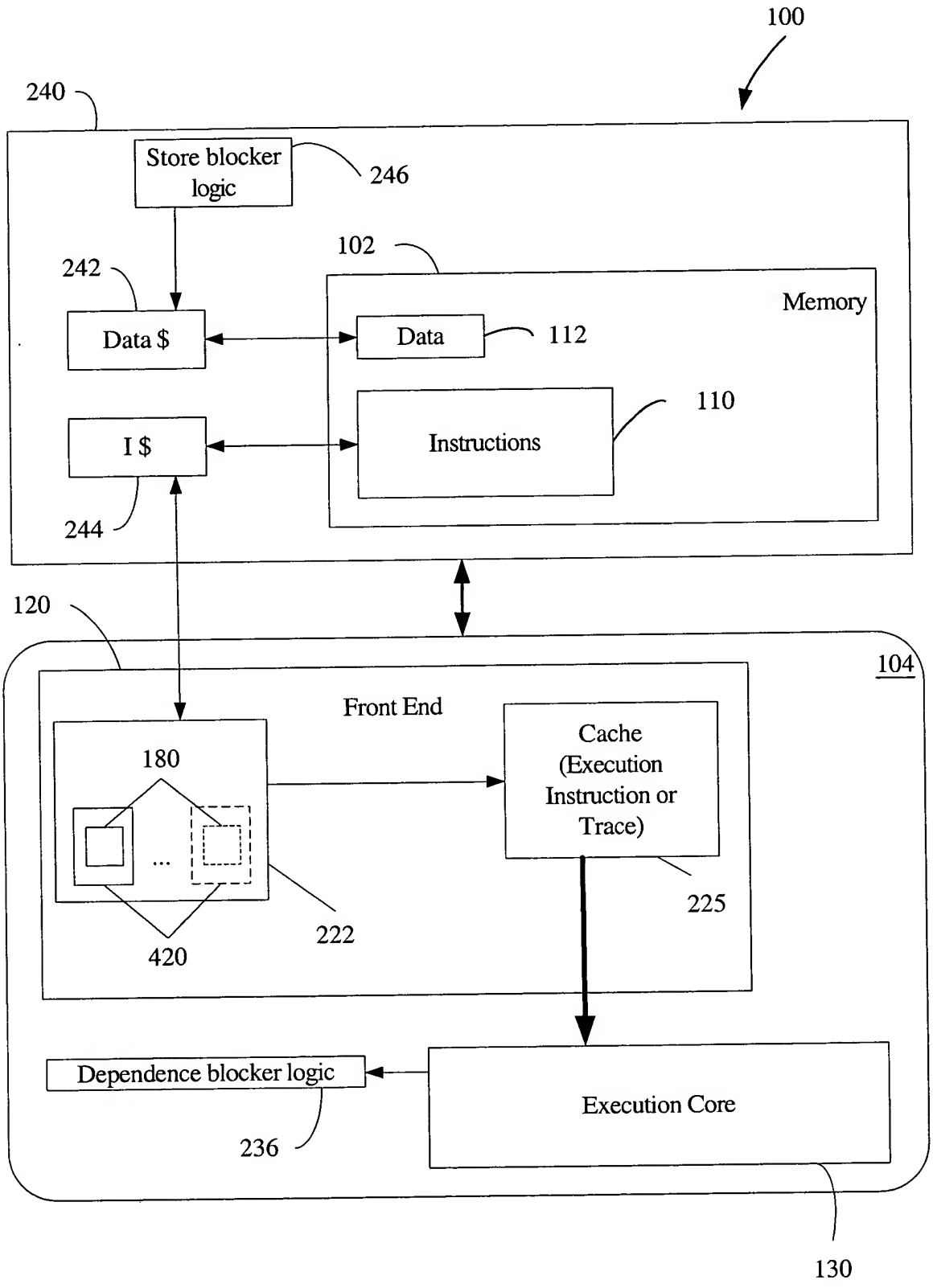
100
240
Store blocker logic
246
102
242
Memory
Data $
Data
112
110
I $
Instructions
244
120
104
Front End
180
Cache (Execution Instruction or Trace)
225
222
420
Dependence blocker logic
Execution Core
236
130

Fig. 1

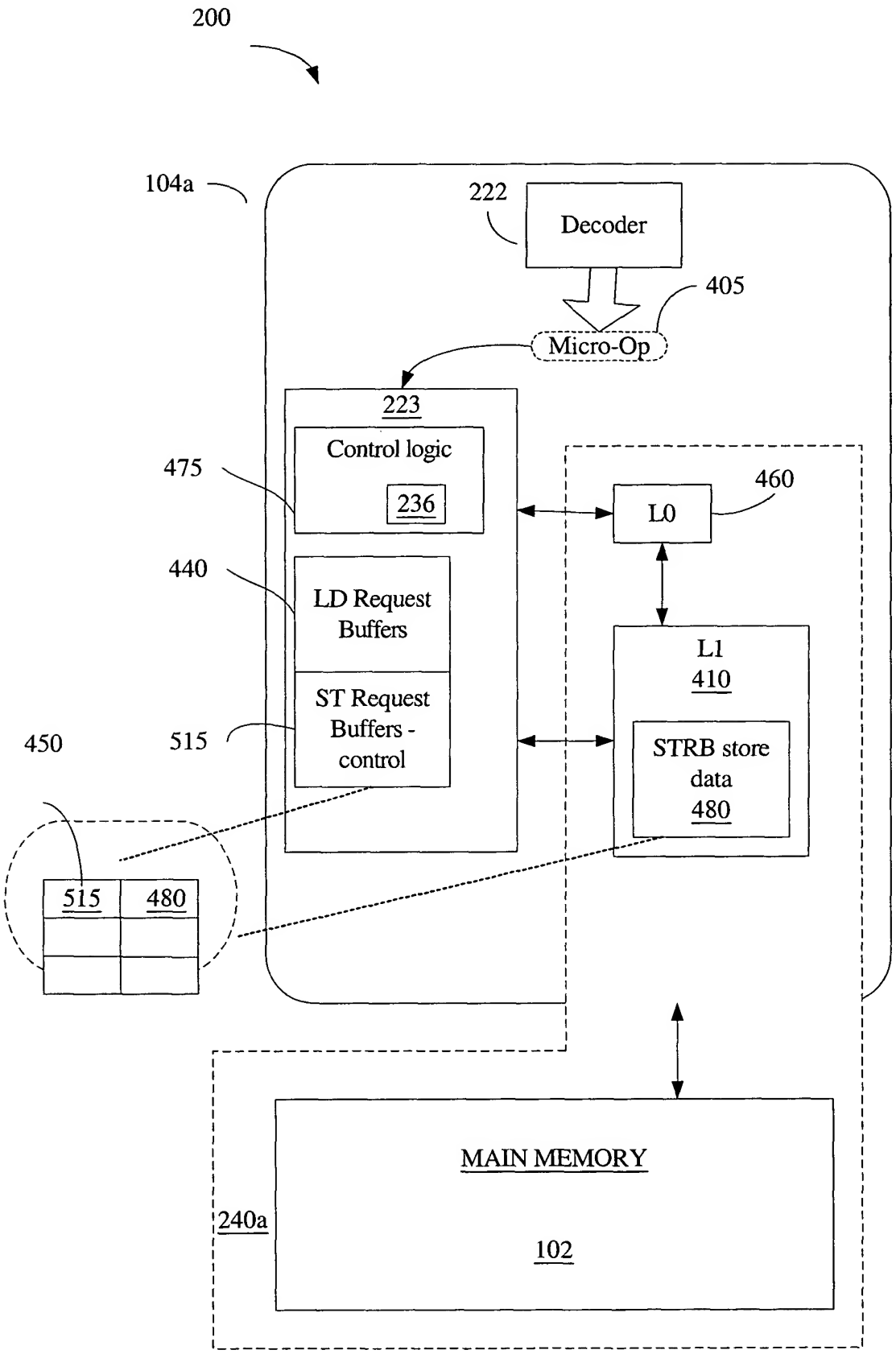
200
104a
222
Decoder
405
Micro-Op
223
Control logic
236
475
460
L0
440
LD Request Buffers
ST Request Buffers - control
515
450
L1
410
STRB store data
480
515
480
MAIN MEMORY
240a
102

Fig. 2

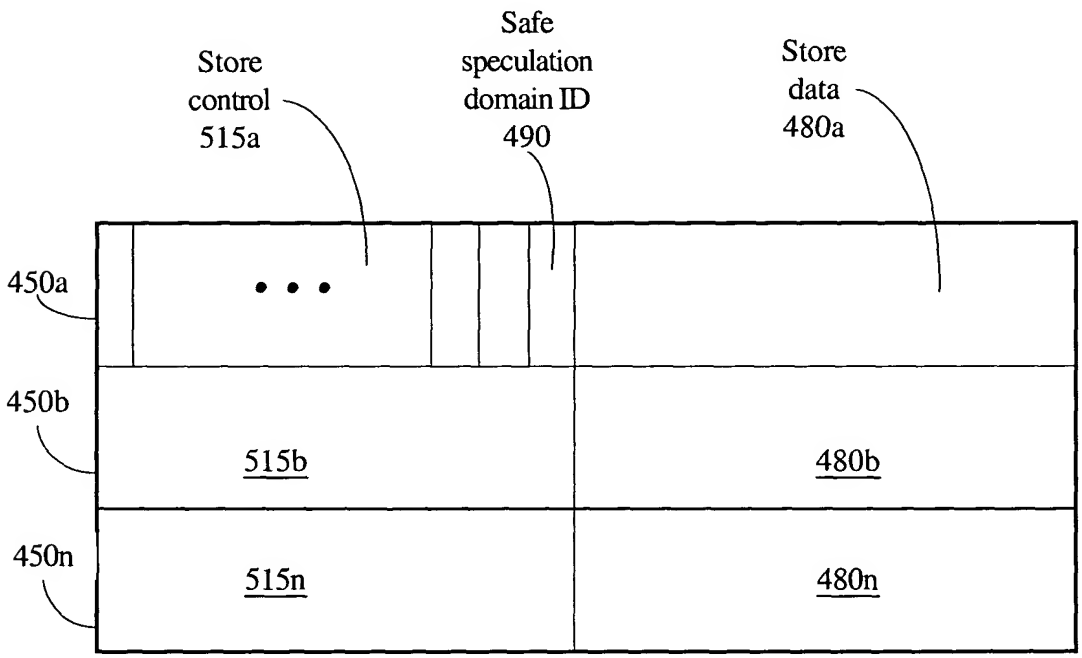
Store control 515a
Safe speculation domain ID 490
Store data 480a
450a
450b
515b
480b
450n
515n
480n

Fig. 3

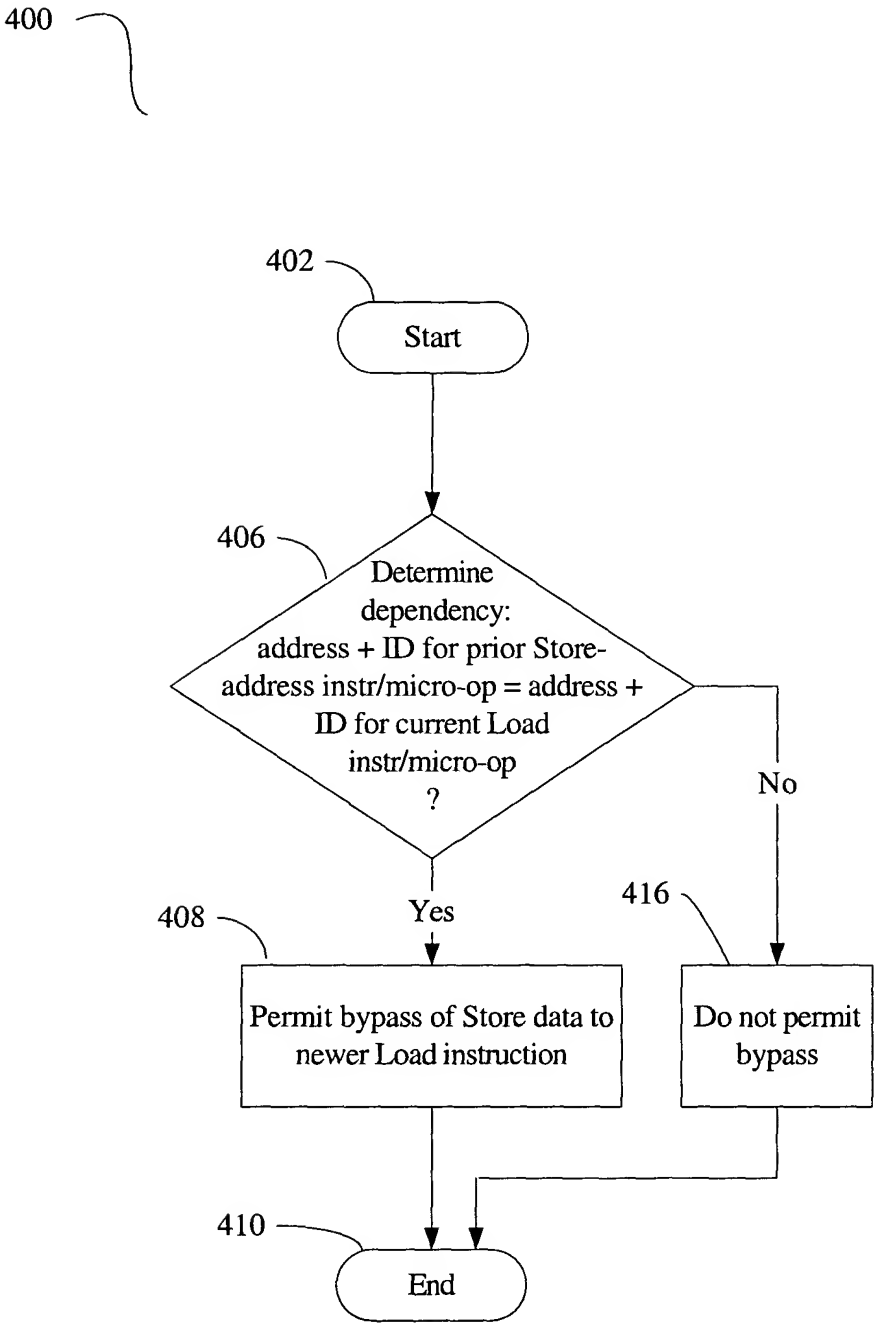
400
402
Start
406
Determine dependency: address + ID for prior Store-address instr/micro-op = address + ID for current Load instr/micro-op ?
No
Yes
408
416
Permit bypass of Store data to newer Load instruction
Do not permit bypass
410
End

Fig. 4

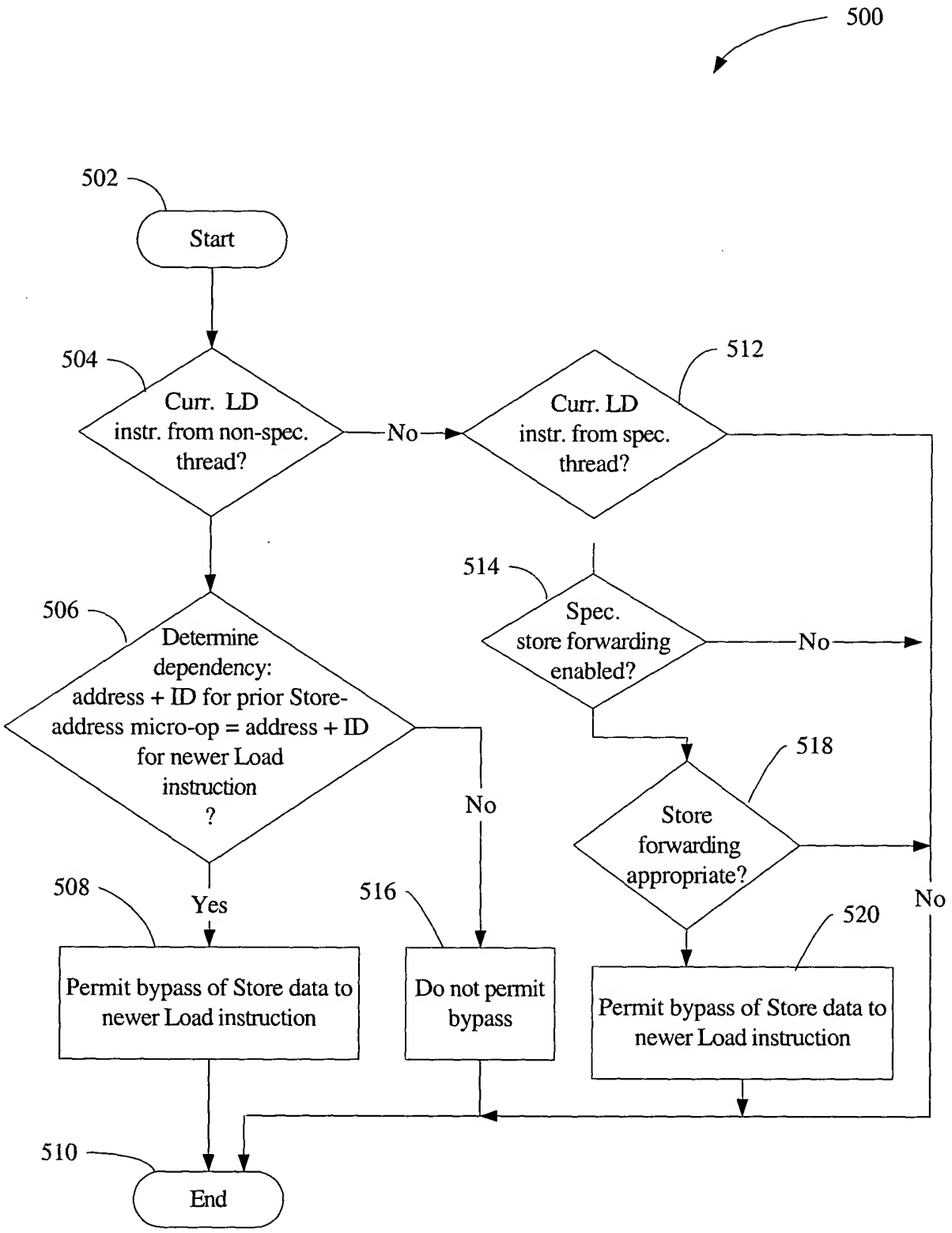
500
502
Start
504
Curr. LD instr. from non-spec. thread?
No
512
Curr. LD instr. from spec. thread?
514
Spec. store forwarding enabled?
No
506
Determine dependency: address + ID for prior Store-address micro-op = address + ID for newer Load instruction ?
No
518
Store forwarding appropriate?
No
Yes
508
Permit bypass of Store data to newer Load instruction
516
Do not permit bypass
520
Permit bypass of Store data to newer Load instruction
510
End

Fig. 5

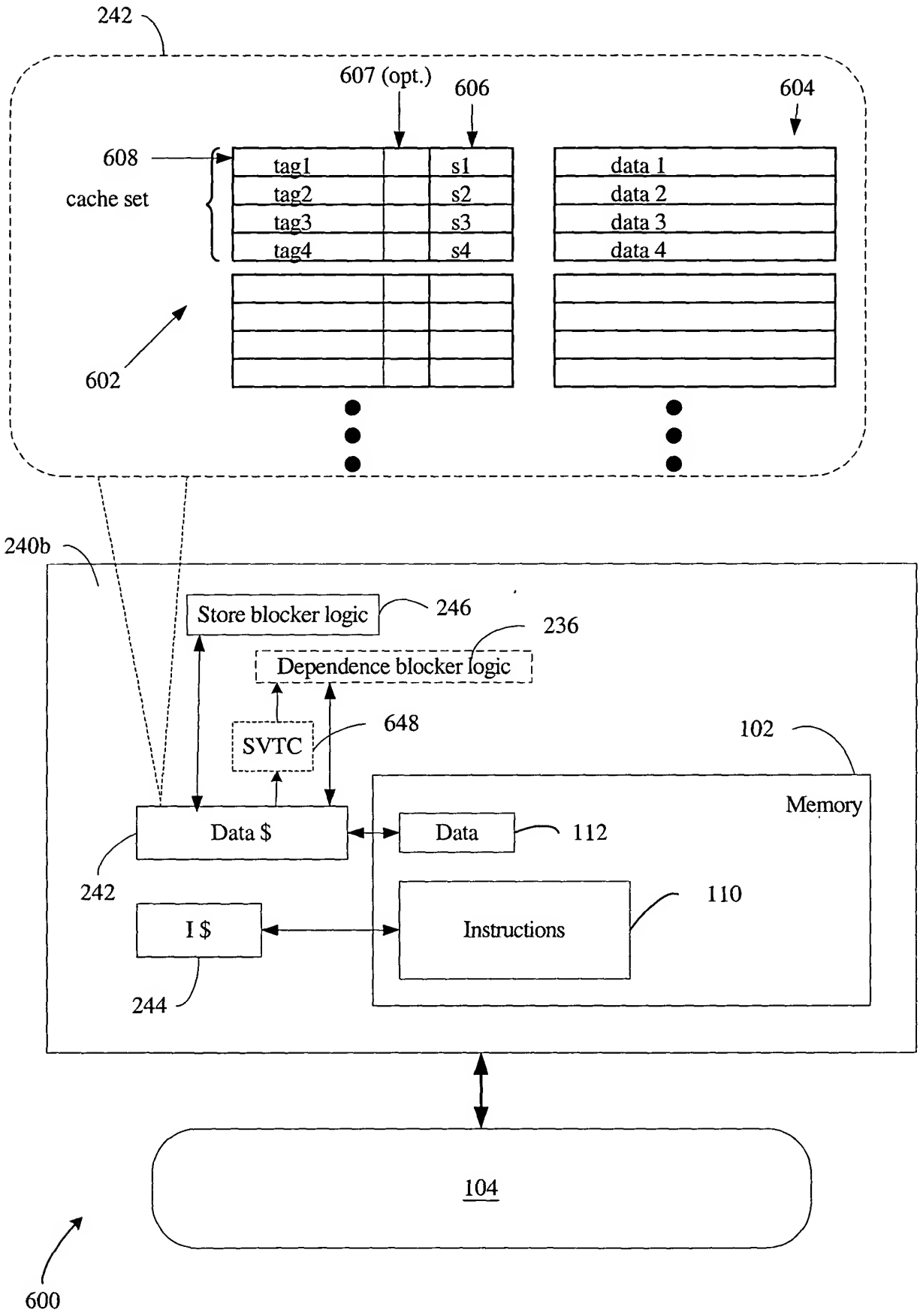
242
607 (opt.)
606
604
608
cache set
tag1
tag2
tag3
tag4
s1
s2
s3
s4
data 1
data 2
data 3
data 4
602
240b
Store blocker logic
246
236
Dependence blocker logic
SVTC
648
102
Memory
Data $
Data
112
242
I $
Instructions
110
244
104
600

Fig. 6

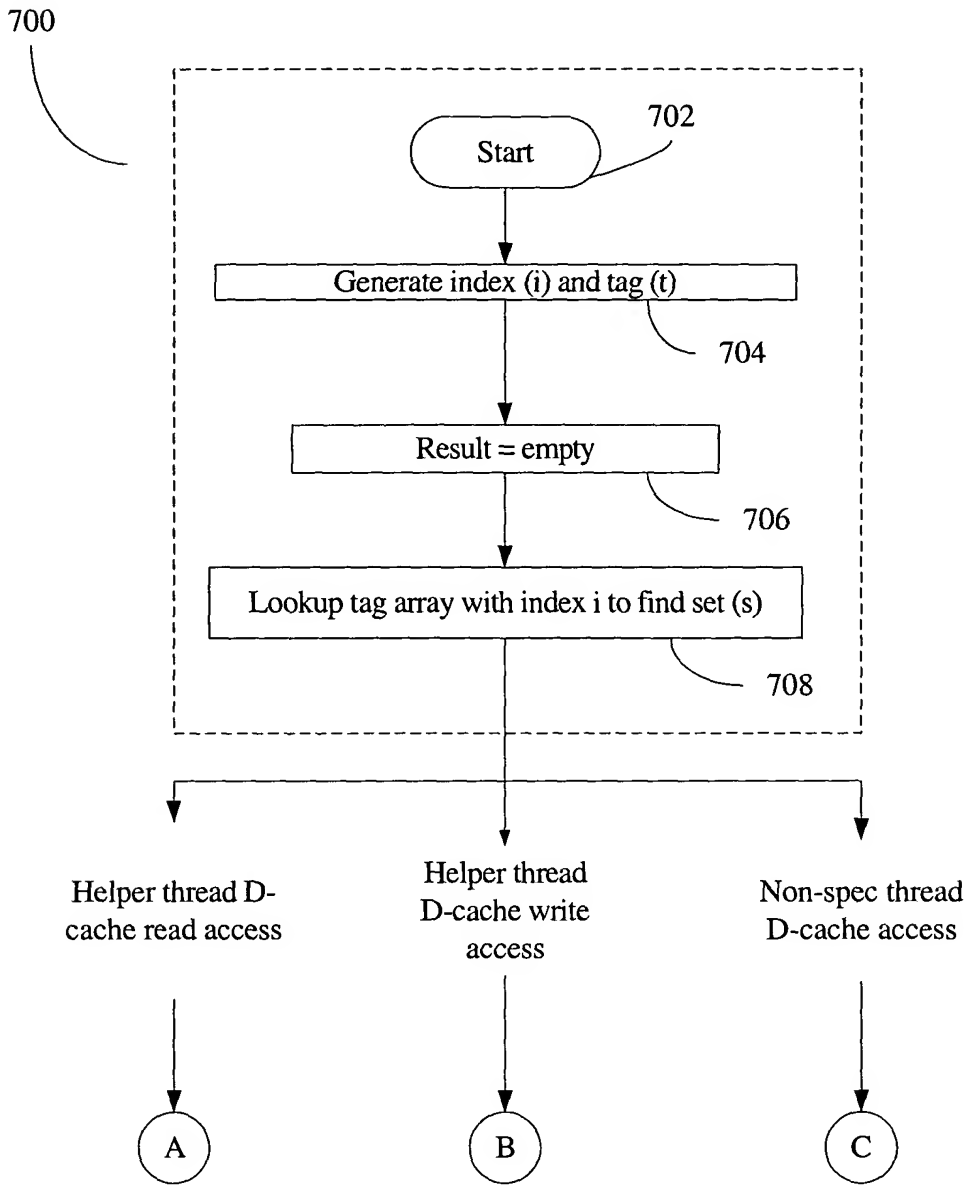
700
Start
702
Generate index (i) and tag (t)
704
Result = empty
706
Lookup tag array with index i to find set (s)
708
Helper thread D-cache read access
Helper thread D-cache write access
Non-spec thread D-cache access
A
B
C

Fig. 7

Helper thread D-cache read access

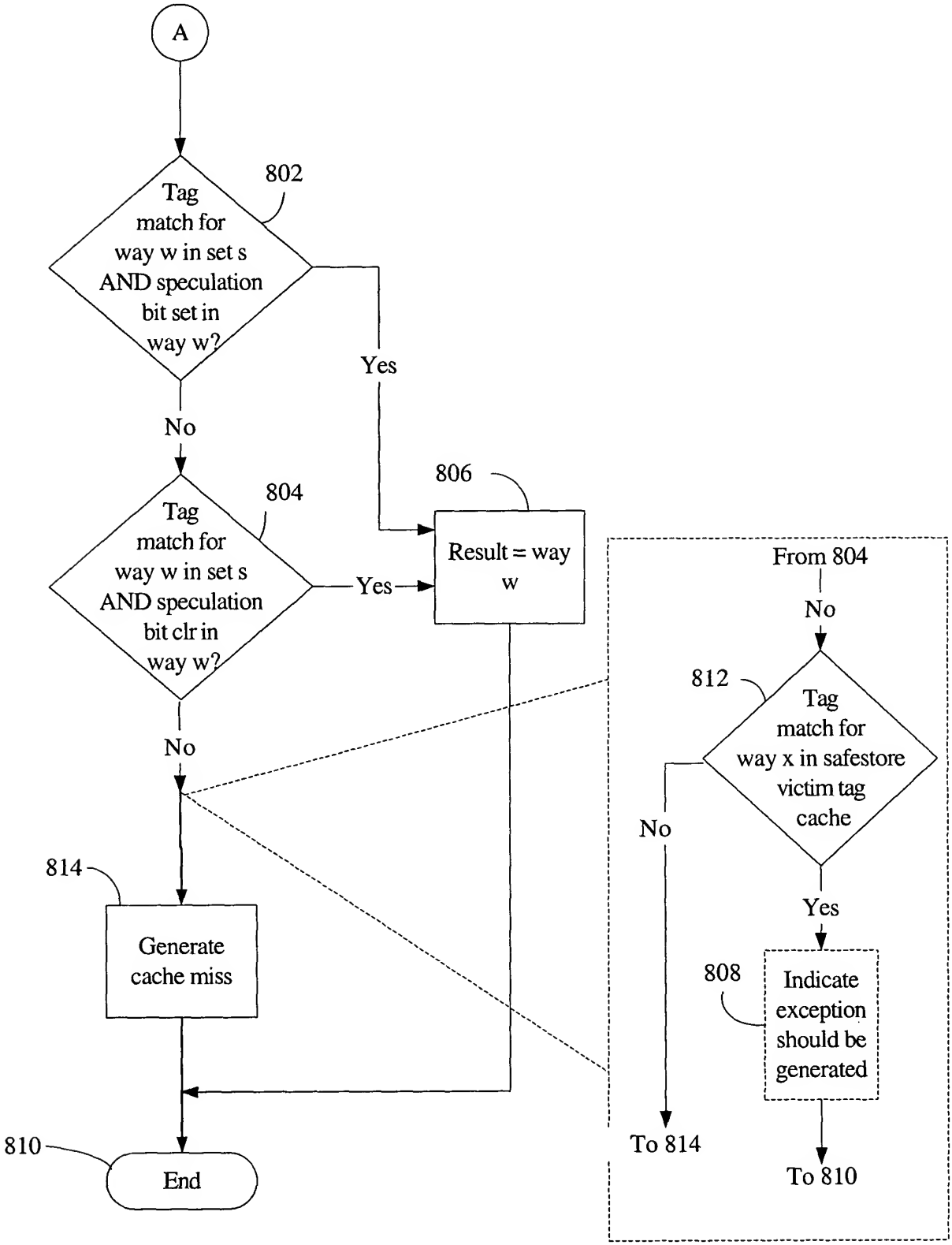

Fig. 8

Helper thread D-cache write access

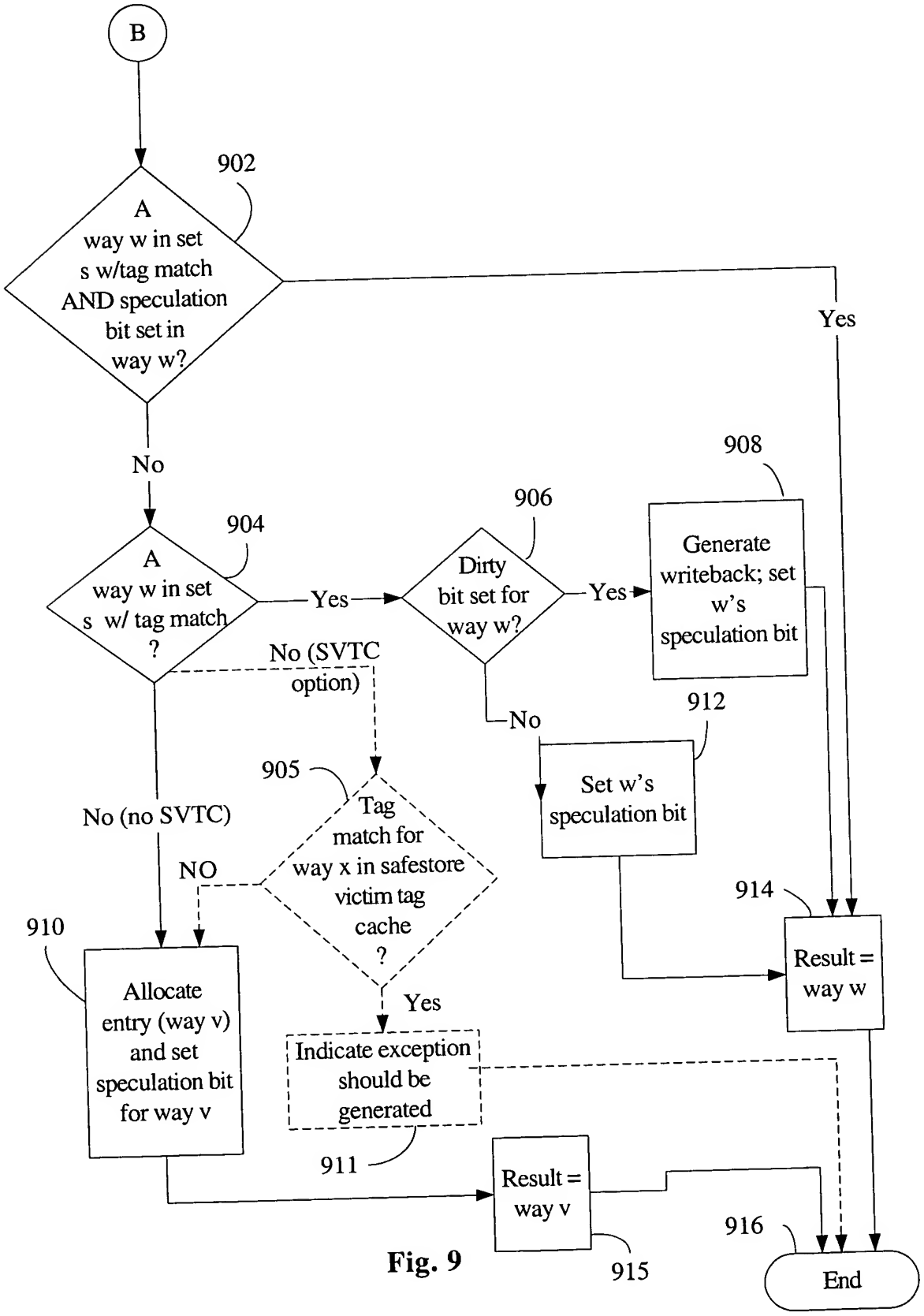

Fig. 9

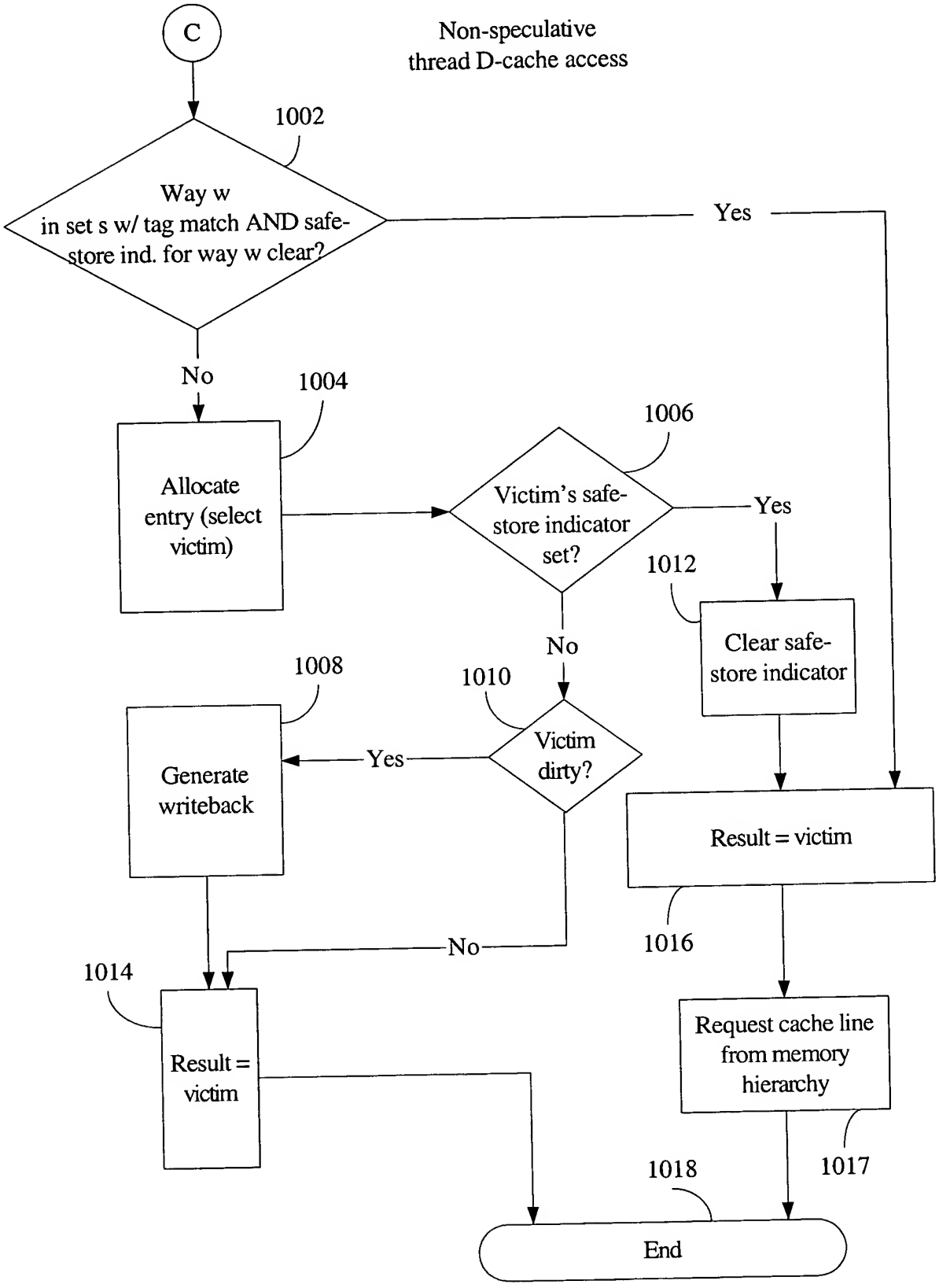
C
Non-speculative thread D-cache access
1002
Way w in set s w/ tag match AND safe-store ind. for way w clear?
Yes
No
1004
Allocate entry (select victim)
1006
Victim's safe-store indicator set?
Yes
1012
Clear safe-store indicator
No
1008
1010
Victim dirty?
Generate writeback
Yes
Result = victim
1016
No
1014
Result = victim
Request cache line from memory hierarchy
1017
1018
End

Fig. 10

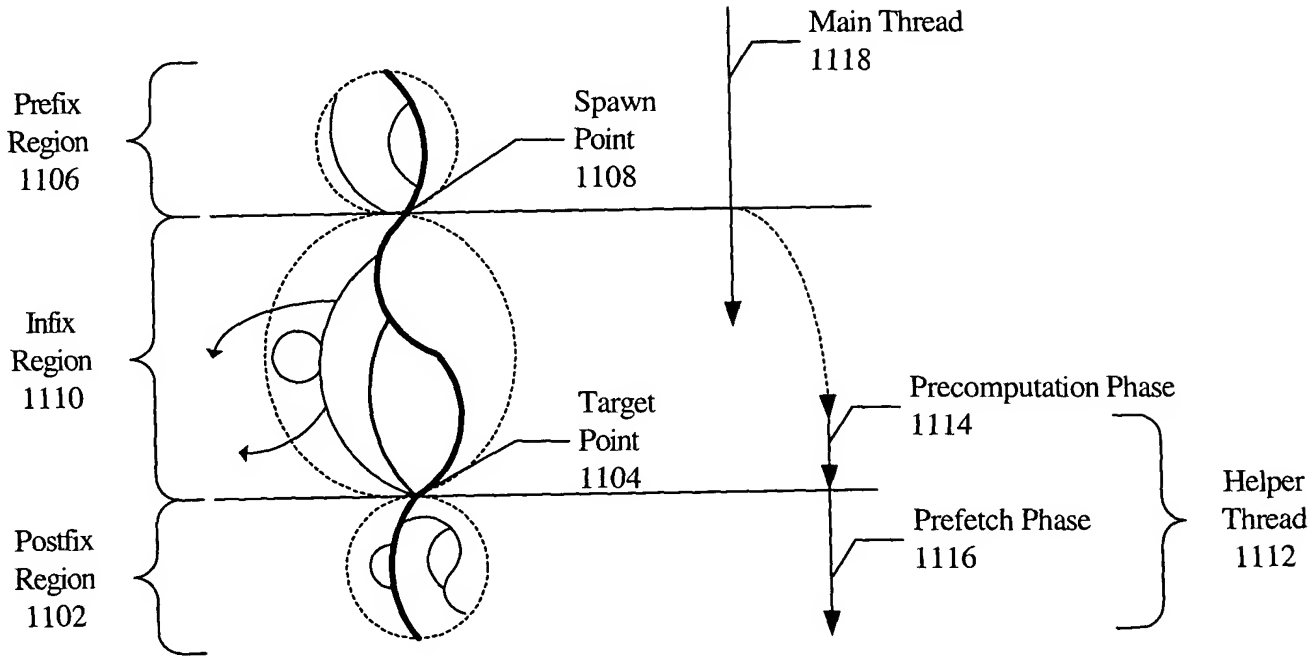
Main Thread
1118
Prefix
Region
1106
Spawn
Point
1108
Infix
Region
1110
Target
Point
1104
Precomputation Phase
1114
Prefetch Phase
1116
Helper
Thread
1112
Postfix
Region
1102

FIG. 11

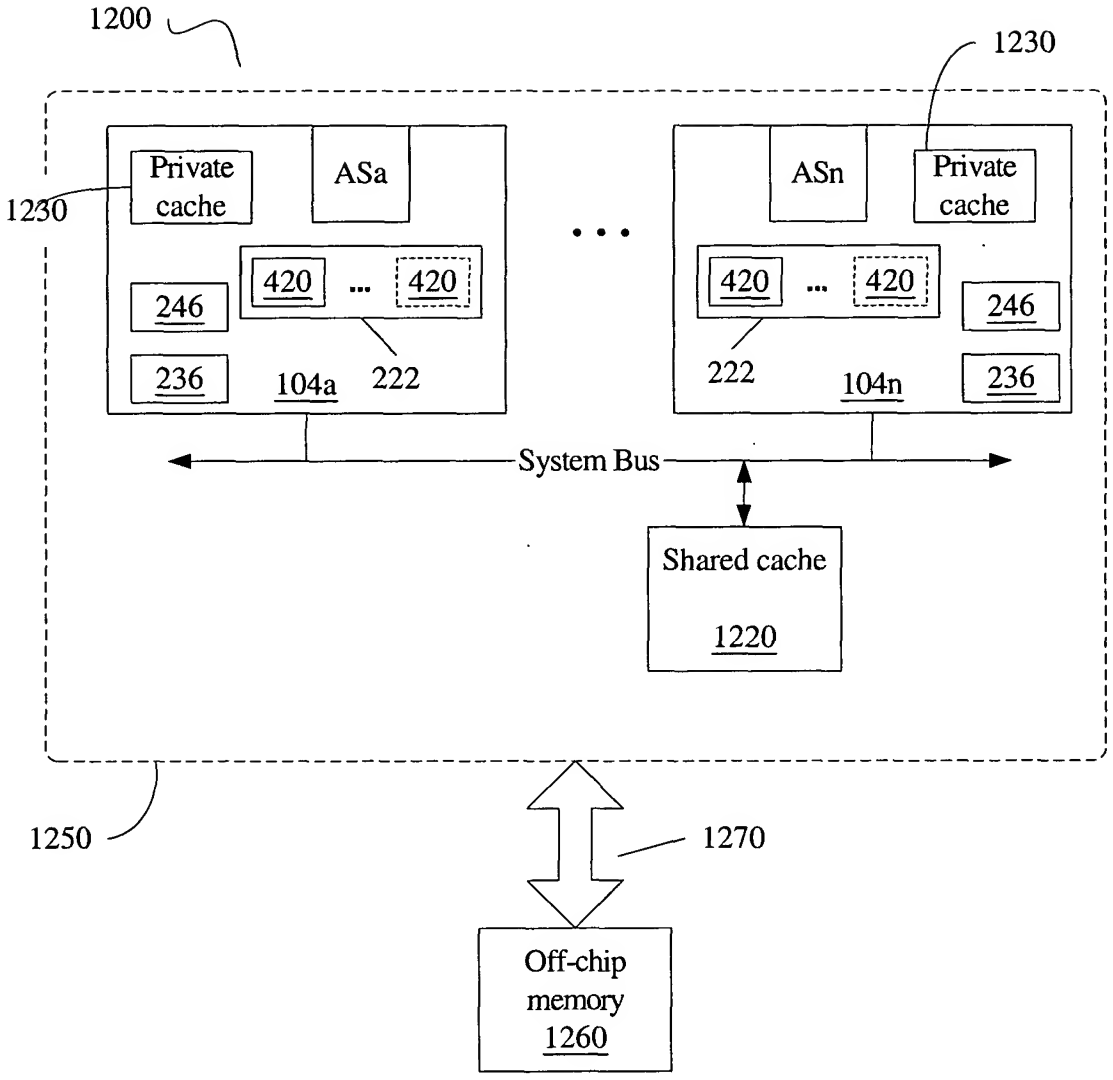
1200
1230
1230
Private cache
ASa
ASn
Private cache
420
420
420
420
246
246
236
236
104a
222
222
104n
System Bus
Shared cache
1220
1250
1270
Off-chip memory
1260

Fig. 12